UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/759,235	01/20/2004	Byoung-Chul Kim	P57009	7206

7590 10/24/2008

Robert E. Bushnell
Suite 300
1522 K Street, N.W.
Washington, DC 20005

EXAMINER	
SHIN, KYUNG H	

ART UNIT	PAPER NUMBER
2443	

MAIL DATE	DELIVERY MODE
10/24/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/759,235	KIM ET AL.	
	Examiner	Art Unit	
	KYUNG H. SHIN	2443	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 January 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-10* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *1/20/04* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *1/20/04, 7/13/06, 8/17/07*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responding to application papers filed on **1-20-2004**.

2. Claims **1 - 10** are pending. Claims **1, 2, 7, 9** are independent.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless -
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claim **1** are rejected under 35 U.S.C. 102 (e) as being anticipated by **Civanlar et al.** (US Patent No. **6,078,963**).

Regarding Claim 1, Civanlar discloses a distributed router comprising:

a) a plurality of routing nodes each having a plurality of routing protocols; (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in a overall network (plurality of routers); col 3, ll 37-41: any known types of routing protocols packets may be received (OSPF, RIP, BGP4)) and

b) a switching module having a plurality of routing protocols corresponding to the routing protocols of each of the routing nodes, disposed to share in real time routing

information collected by each of the routing nodes with others of the routing nodes. (Civanlar col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database; col 3, ll 37-41: any known types of routing protocols packets may be received (OSPF, RIP, BGP4))

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims **2 - 10** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Civanlar** in view of **Venkatachary et al.** (US Patent No. **6,212,184**).

Regarding Claim 2, Civanlar discloses a method of managing forwarding information, comprising the steps of:

when new routing information is inserted into a routing table in a distributed router in which all routing nodes share a forwarding information. (Civanlar col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses:

(1) wherein an aggregation tree based on the routing table, detecting a position at which an insertion node corresponding to the new routing information is to be inserted into the aggregation tree; (Venkatachary col. 15, lines 50-60: search and update (insert) routing information)

(2) determining presence and absence of an ancestor node of the insertion node at or below a predetermined maximum aggregation level; (Venkatachary col. 10, lines 6-33; col 16, ll 26-36: determine existence ancestor node)

(3) leaving the forwarding table un-updated with information about the insertion node in a presence of the ancestor node, when forwarding information is in the forwarding table and the insertion node and the ancestor node have been generated from a common source area; (Venkatachary col. 16, lines 37-52: switch pointer inserted when nil; no node exists)

(4) in an absence of the ancestor node, resetting the aggregation level to a reset aggregation level not greater than the maximum aggregation level, and inserting a delegation node representative of the insertion node at the reset aggregation level; (Venkatachary col. 16, lines 26-36: switch pointer; reset to ancestor node level) and

(5) making an insertion of forwarding information by determining the source area of the inserted routing information, inserting forwarding information corresponding to the delegation node in the forwarding table when the source area of the routing information is a virtual area, and inserting forwarding information

corresponding to the insertion node in the forwarding table when the source area of the routing information is a local area. (Venkatachary col. 15, lines 50-60: update (insert) forwarding information)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55: *" … It would therefore be advantageous to provide routers and routing methods that did not require either huge memory requirements or large lookup times. … "*)

Regarding Claim 3, Civanlar discloses the method of claim 2. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database) Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein comprised of, after making said insertion of forwarding information when a delegation node is found to exist at the position of the insertion node while detecting a position at which an insertion node corresponding to the new routing information is to be inserted into the aggregation tree, deleting from the forwarding table forwarding information corresponding to the delegation node. (Venkatachary col. 15, lines 50-60: update (insert, delete) forwarding information)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Regarding Claim 4, Civanlar discloses the method of claim 2. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein comprised of: a) after making said insertion of forwarding information when a delegation node is found to exist at the position of the insertion node while detecting said position at which an insertion node corresponding to the new routing information is to be inserted into the aggregation tree, and when a left/right subtree of the delegation node exists, reinserting nodes of the left/right subtree, and deleting forwarding information corresponding to the delegation node from the forwarding table. (Venkatachary col. 10, lines 6-33; col 16, ll 26-36: search forwarding information; col. 15, lines 50-60: update (insert, delete) forwarding information)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of

ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Regarding Claim 5, Civanlar discloses the method of claim 2. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein comprising the steps:

when the ancestor node of the insertion node is found to exist at or below the maximum aggregation level while determining said presence and absence of the ancestor node, searching for a descendant node of the insertion node; (Venkatachary col. 10, lines 6-33; col 16, ll 26-36: search tree; identify and locate ancestor node)

when a descendant node of the insertion node is found to exist, resetting the aggregation level according to a difference between the prefixes of forwarding information corresponding to the insertion node and the descendant node, and when no descendant nodes of the insertion node are found to exist, resetting the aggregation level according to the aggregation level of the ancestor node of the insertion node; (Venkatachary col. 16, lines 26-36: switch pointer; reset and try different branch)

inserting the forwarding information corresponding to the insertion node in the forwarding table when the reset aggregation level is zero; inserting the delegation node representative of the insertion node in the forwarding table when the reset aggregation level is greater than zero; and determining the source area of the inserted routing information, inserting the forwarding information corresponding to the delegation node in the forwarding table when the source area is a virtual area, and inserting the forwarding information corresponding to the insertion node in the forwarding table when the source area is a local area. (Venkatachary col. 15, lines 50-60: search and update (insert) routing information; col. 16, lines 26-36: reset and try different branch)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Regarding Claim 6, Civanlar discloses the method of claim 2. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein comprised of performing said steps of

resetting the aggregation level to a reset aggregation level not greater than the maximum aggregation level in an absence of the ancestor node, and inserting a delegation node representative of the insertion node at the reset aggregation level, by: setting a search level range whether the ancestor node of the insertion node exists within the search level range; when the ancestor node of the insertion node exists within the search level range, determining whether a descendant node of the deletion node representative of the insertion node exists at the maximum aggregation level; (Venkatachary col. 16, lines 26-36: switch pointer; reset and try different branch)

resetting the aggregation level according to a difference between the prefixes of the insertion and the descendant node of the delegation node when the descendant node of the delegation node exists at the maximum aggregation level; and inserting the delegation node of the insertion node at the reset aggregation level. (Venkatachary col. 16, lines 26-36: switch pointer; reset and try different branch)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Regarding Claim 7, Civanlar discloses a method of managing forwarding information. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein comprising the steps:

routing information is deleted from the routing table a deletion node corresponding to the deleted routing information in the aggregation tree; (Venkatachary col. 15, lines 50-60: search and update (insert) forwarding information)

forwarding information corresponding to the deletion node is in a forwarding table, searching for a descendant node of the deletion node at a predetermined maximum aggregation level; (Venkatachary col 10, ll 6-33; col 16, ll 26-36: search for a descendent (ancestor) node) and

a descendant node exists for the deletion node at an aggregation level not greater than a predetermined maximum aggregation level, the descendant node as a new source node of a delegation node, and no descendant nodes exist for the deletion node at an aggregation level not greater than a predetermined maximum aggregation level, forwarding information corresponding to the deletion node from the forwarding table. (Venkatachary col. 15, lines 50-60: search and update (delete) forwarding information)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Regarding Claim 8, Civanlar discloses the method of claim 7. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein comprising the step of, the deletion node is a source node that created a delegation node, forwarding information corresponding to the delegation node forwarding information corresponding to the deletion node. (Venkatachary col. 15, lines 50-60: search and update (delete) forwarding information)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Regarding Claim 9, Civanlar discloses a distributed architecture router, comprising:

a switching module accommodating a plurality of routing protocols while managing forwarding information within the distributed architecture router; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database; col 3, ll 37-41: any known types of routing protocols packets may be received (OSPF, RIP, BGP4)) and

a plurality of routing nodes each disposed to service networks within corresponding source areas comprised of local areas, said plurality of routing nodes being connected via said switching module to form a source area comprising a virtual area and share in real time collected routing information assembled by a routing table. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in a overall network; col 3, ll 37-41: any known types of routing protocols packets may be received (OSPF, RIP, BGP4))

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein an aggregation tree derived from routing table information. (Venkatachary col 9, l 64 - col 10, l 5: search a tree for a lowest cost match and routing the packet)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of

Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 521-55)

Regarding Claim 10, Civanlar discloses the distributed architecture router of claim 9, comprised of said routing nodes responding to insertion of new routing information into said routing table. (Civanlar col. 2, lines 53-55: coupled to network nodes such as routers/switches in an overall network; col. 3, lines 41-47: forwarding engine configured to forward new routing table configuration data to every other router port for updating database)

Civanlar does not explicitly disclose an aggregation tree for routing information. However, Venkatachary discloses wherein:

identifying in said aggregation tree a position for addition of an insertion node corresponding to said new routing information; (Venkatachary col. 15, lines 50-60: search and update (insert) forwarding information)

making a search of said aggregation tree within a maximum aggregation level to identify an ancestor node of said insertion node; (Venkatachary col. 10, lines 6-33; col 16, ll 26-36: search tree; identify and locate ancestor node)

forgoing updating of said forwarding table with forwarding information corresponding to said insertion node when said insertion node and said ancestor node were generated the same said source area and said search identifies said ancestor node; (Venkatachary col. 16, lines 37-52: no update; no node at location)

resetting said maximum aggregation level to a reset aggregation level not less than said maximum aggregation level when said search fails to identify said ancestor node and adding a delegation node representative of said insertion node at said reset aggregation level; (Venkatachary col. 16, lines 26-36: switch pointer; reset and try different branch)

making an identification of said source area of said new routing information; inserting said forwarding information corresponding to said delegation node when said identification establishes that said source area of said new routing information is a virtual area; (Venkatachary col. 15, lines 50-60: search and update (insert) forwarding information) and

inserting said forwarding information corresponding to said delegation node when said identification establishes that said source area of said new routing information is a local area. (Venkatachary col. 15, lines 50-60: search and update (insert) forwarding information)

It would have been obvious to one of ordinary skill in the art to modify Civanlar to utilize an aggregation tree for routing information as taught by Venkatachary. One of ordinary skill in the art would have been motivated to employ the teachings of Venkatachary in order to provide routers and routing methods that did not require either huge memory requirements or large lookup times. (Venkatachary col 9, ll 52-55)

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to KYUNG H. SHIN whose telephone number is (571)272-3920. The examiner can normally be reached on 9:30 am - 6 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Nathan J. FLYNN can be reached on (571) 272-1915. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Kyung Hye Shin
Examiner
Art Unit 2443

KHS
October 13, 2008

/Tonia L. M. Dollinger/

Supervisory Patent Examiner 2443